



The Adoption of Agile Processes in Large Web Development Enterprises: A Survey in Jordan



Omaima N. A. AL-Allaf
Department of Computer Information Systems
Faculty of Sciences and Information Technology
AL-Zaytoonah Private University of Jordan
P.O. Box 130, Amman (11733), Jordan
omaimaalallaf@alZaytoonah.edu.jo

ABSTRACT: Agile processes can be applied to any software process for effective response to changing requirements during the development process and effective communication among stakeholders. There is lack of surveys related to agile processes in large web development enterprises in the world and the current state of adoption of agile processes by large enterprises is still unknown. A survey has been conducted in this research based on questionnaire and interviews in five large Jordanian enterprises to examine the process models adopted by these enterprises, the extent to which these enterprises are using agile processes, and finally to examine the problems associated with large web applications' development. The results of this survey showed that there is a weakness in adoption of agile processes in large Jordanian enterprises and the Extreme Programming (XP) is well known agile process by the developers working in these enterprises. According to survey results, we suggested recommendations related to adoption of XP agile process in large enterprises.

Keywords: Agile software development, Web engineering, Large enterprises, Large web applications, Extreme Programming (XP)

Received: 1 March 2010, Revised 14 April 2010, Accepted 22 April 2010

© 2009 D-Line. All rights reserved

1. Introduction

The web applications vary in scope and complexity from small to large-scale applications distributed across internet [1]. There are many quality criteria for the success of web applications such as usability, security, availability, scalability and maintainability [2]. Web engineering is multidisciplinary field and encompasses inputs from diverse areas such as human-computer interaction, user interface, software requirements, hypermedia engineering, testing, management, modeling and simulation, social sciences, arts and graphic design [3].

The developers of web applications need an iterative process to: manage the development process in a systematic manner and delivers web applications within time [4]. Organizations of web development face unique challenges based on rapidly changing markets and poorly-defined requirements. Different organizations may use different processes to produce the same type of software. Some processes are more suitable than others for some types of software. If an inappropriate process is used, this will reduce the quality of software to be developed [5][6].

1.1 Agile Processes for Web Development

Web applications are constantly changing, have short development cycles and built by a small multidisciplinary team (3 to 10 people) [7]. Web applications present important characteristics that must be addressed by software processes and these applications have to be delivered for the clients within time constraints [8]. The organizations continue to move toward the development of web applications and they assign such projects to small teams of qualified developers. Web applications demand faster time to market and continual integration of new requirements. Therefore, such demands have increased the popularity of agile processes which let teams increase productivity and increase organization's responsiveness. The XP focuses on small teams and replaces paper-based documentation with face-to-face communication. As the development organization

grows, the time spent in exchanging product knowledge and training new people increases and often renders XP unsuitable [9]. Adopting agile in web development is a process of continuous learning and improvement. The transition to agile requires hard work to successfully deliver quality web applications and improve quality of team members. The organization should begin working on team building and learning how to work better across the teams [10].

Grossman, et. al. [11] suggested that, the XP make sense as a development method in multidisciplinary and diverse web development environment by adopting all of XP practices. This environment includes diverse and distributed teams, requires close coordination with multidisciplinary skills to make development process more responsive to changing requirements, decreases cost and time and increases user satisfaction. Sampaio, et. al. [8] described an agile process for web development (XWebProcess) based on XP. The XWebProcess used for building quality web applications in effective way and suitable to web development in dimensions such as requirements gathering, user interface design and testing.

1.2 Large Web Applications

The simplest form of web applications consists of a set of linked hypertext files that present information. When the importance of e-business applications grow, the web applications are evolving into sophisticated computing environments that provide functions to end user, and are integrated with databases [5]. The large web application has many properties: include large quantities of code; include large number of interactive functions; include hundreds to thousands pages of documentation; use of multiple programming languages; and developed by large number of developers (more than 50) [12][13][14]. Building large web applications are difficult, costly, and time-consuming and faced system and network challenges [15]. The development of large web applications should be carried out through planning and design method, and by integrating management and web technologies [16].

1.3 The Importance of this Research

The software process is a technical framework established for applying tools, methods and people to the software task. The organizations need a defined process to provide them with a consistent framework for performing and improving their work. When several people work on a common project, they need some way to coordinate their work. For small tasks, this can be done informally, but with larger numbers of people or more complex tasks, more formal arrangements are needed [17]. Dorothy [18] addressed the problems of large web development such as: it is difficult to estimate the cost, effort, time or size of large applications; and changing requirements during the development process. Many web development methods focused on user interface design but failed to address the overall development process [19]. Traditional process models have challenges to accommodate web specific aspects into their techniques. Therefore, web development requires a mix of web development techniques with properties of traditional process models [20].

The survey is a popular research tool often used in empirical software engineering studies. Surveys should be promoted to gather information about what software engineers actually do, and to evaluate prescribed practices, methods, tools and standards. Survey may be administered by researcher, or distributed by post, or via internet [21]. There is lack of surveys related to agile processes in large web development enterprises in the world and also the current state of agile processes' adoption by large enterprises is still unknown.

To understand the level of adoption of agile processes currently in use in large enterprises, a survey has been conducted in this research based on questionnaire and interviews in five large Jordanian enterprises. The objective of this survey is to examine software development models adopted by these enterprises, the extent to which these enterprises are adopting agile processes, and finally to examine the problems associated with web development in these enterprises. The benefits of such survey is to improve the software processes for large web development used by these enterprises by detecting the level of adoption of software processes including agile processes, also by detecting the problems during the development of web applications. The remaining of this research is as follows: section 2 includes literature review, section 3 includes research methodology and section 4 includes statistical analysis. Section 5 describes mixing agile with traditional processes and section 6 includes recommendations. Section 7 includes evaluation and finally, section 8 concludes this paper.

2. Literature Review

Many software process models were developed to support the development process. The most well known used traditional processes are Waterfall, Incremental, Spiral, Prototyping and Rational Unified Process (RUP). Each model has its own approach, strengths and limitations [5]. There are many problems of traditional process models. The most frequent problem of these methods is that they are bureaucratic. The several phases in development slow down the development process. The second problem is that the requirements specifications are not flexible since it is difficult to get the customers to identify their requirements. Even if the requirements can be identified, the business world is forever changing [5][19].

Agile processes are approaches to build systems that emphasize evolutionary development, customer-centricity and low-documentation overhead [22]. Agile Manifesto establishes a common framework for these processes [23]: “individuals and interactions over processes and tools; working software over comprehensive documentation; customer collaboration; and responding to change”. A software process is considered agile when it conforms to these above four values and principles [24]. The 12 principles introduced behind agile manifesto are: customer satisfaction; welcome changing requirements; deliver working software frequently; business people and developers must work together daily; build projects around motivated individuals; face-to-face conversation within team; working software is the measure of progress; agile processes promote sustainable development; continuous attention to technical excellence; simplicity; self-organizing teams; and effectiveness [23]. There are many various agile processes such as: agile manifesto, XP, scrum, crystal, feature-driven development (FDD), dynamic system development method (DSDM), agile modeling (AM), and adaptive software development (ASD) method [24]. XP is the most widely used agile development method and it is one of the most discussed subjects in development community [25]. It is proposed by Kent [26] and aim to reduce the cost of change by introducing basic values, principles and practices [27].

XP describes four activities: listening; planning; designing; coding; and testing that are performed within the development process [26]. The proper use of agile processes requires an understanding of the situations in which agile processes are and are not applicable [28]. The industry of information technology has not seen large scale adoption of agile methods [29] because there are many shortcomings of agile processes such as limited support for: distributed development environments; development involving large teams; developing safety-critical software; and developing large software [28][29]. The benefits of agile methods are: shortened development cycle-time by 75%; higher stability of work-loads; higher flexibility to change of development plans; and higher quality by earlier feedback from customers [30].

2.1. Agile Processes in Large Enterprises

Agile processes allow changing requirements during the development cycle and stress collaboration between developers and customers. The transition from traditional to agile processes affects the development team members and management. Effective approaches should be taken to make this change [31][32]. The development teams range in size from two to several individuals in organizations. Therefore, the development process that is appropriate for very large teams will not work well for small teams. Small teams can be flexible and adaptable in applying agile methods. For a team with size that exceed 10, the large team can be divided into collections of smaller independent sub teams, each no larger than 10 and define interfaces between them [31].

The success of agile processes in large organizations requires teams of skill people to be effective and trusted. The agile processes do not work well for systems that have criticality, reliability and safety requirements. The agile processes are more appropriate when requirements are rapidly changing because these processes are based on close interaction with the customers. The organizations should analyze their past projects from the perspective of success factors such as culture, people and communication. To be agile is a cultural thing and the organization cannot be agile if its culture is not right. The teams need amount of local control and must have the ability to adapt working practices as they feel appropriate.

Organizations that want to be agile need to have an environment that facilitates rapid communication between team members [33]. Boehm [34] suggested that, the most important factor that should be determined when agile is applicable is the project size. Agile processes are more difficult for larger teams. At the same time, traditional processes do not fit the small projects and these processes are most needed in high-assurance software [34]. Finally, Khalaf and Al-Amawi [35] studied the differences that inferred by using Extreme programming (XP) in a set of software development companies in Jordan in 2008. The analysis of their questionnaire showed that developers who use XP, are focusing on software quality (SQ) more than others.

3. Research Methodology

The analysis units for this survey are five large Jordanian enterprises which undertaken the development of large web applications. The questionnaire's instrument has three parts: the first part related to enterprises' characteristics such as number of developers, application domain, test types, and assurance activities. The second part related to developers' familiarity with software processes including agile processes. The third part related to software processes used by these enterprises including agile processes.

This questionnaire was written in English language and reviewed and validated many times by four professional professors currently working in different Jordanian Universities and specialized in software engineering and management. The selection of these large enterprises was based on definitions of large projects mentioned in researches [5][12][13][14][15]. At the same time, the size of enterprises in Jordan is determined by the Statistical Office according to many factors such as: number of

employees, budget, size of projects, and the time required for development process. The selected five enterprises have many characteristics such as they: involve 50 and more developers in development process; use more than three programming languages in development; develop web applications to provide 50 and more functions, and more than 100 web pages to users and with more than hundreds of thousands of lines of code; project duration ranging from one to three years; finally many of these enterprises have many branches in other countries.

A two of statistical techniques (descriptive statistics and explanatory factor analysis) were utilized in data analysis. Descriptive statistics (frequencies and percentage) were used to identify the major characteristics of large enterprises, developers' familiarity with development processes, and the development processes used by these enterprises. The descriptive statistics and multivariate statistical were used in term of description of problems in these enterprises.

The research sample consisted of Two hundred and Fifty developers working in these five large enterprises in Jordan. The determination of: the number "250"; the distribution of different number of questionnaires; and the selection of invitees are done according to discussion with the Human Resource Department and Manager in each one of the five enterprises. Table (1) describes the number of distributed and retrieved questionnaire in each enterprise. The number of retrieved valid questionnaires is One hundred and Eighty, the reason of why the other 70 developers were not answer is that; many of them haven't enough time to fill the questionnaire; or they were outside Jordan (working, training or holiday) at survey time. These one hundred and Eighty questionnaires were used later in this research in the statistical analysis.

Org. #	No. of developers	No. of Distributed Questionnaires	No. of retrieved Questionnaires
1	110	60	48
2	100	60	40
3	85	45	33
4	78	45	31
5	65	40	28
Total		250	180

Table 1. Research Samples

3.1 Interviews

Interviews were carried out with developers working in these selected five enterprises. The objective of these interviews is to investigate the problems associated with time, cost and effort estimation, the success rate of web applications development, and finally the reasons for web project failure. The descriptive statistics and multivariate statistical were used in term of description of problems in these enterprises.

The statistical analysis related to interviews in all selected enterprises showed that: many of these enterprises suffer from many problems: requirements are always changes during the development process; poor of communication between customers and developers; late customer feedback; lack of standardization; lack of refactoring; poor project management; poor project cost, time, and effort estimation techniques; communication cap between the developers in different teams; lack of professional developers; and problems of deadlines.

4. Statistical Analysis

This section includes the descriptive statistics for the answers of the questionnaire.

4.1. Characteristics of Large Enterprises

The results related to the different sizes of large Jordanian enterprises showed that all of these enterprises have 60 or more developers.

The results related to application domain in these enterprises showed that the business information systems got the highest percentage 66%, whereas e-business got percentage 50%, e-banking got percentage 40%, e-learning got percentage 31% and finally, personal web pages got percentage 23%. The results related to "if the web application development involved in-house, outsourcing, or reusability?" showed that the "in-house development" got the highest percentage 99%. The "outsourcing" got percentage 42% and finally the "reusability" got percentage 23%.

The results related to test types performed by these enterprises showed that: integration test got the highest percentage 64%, web metrics got percentage 30%, unit tests got percentage 27%, database test got percentage 17%, and performance test got percentage 6%.

At the same time, the results related to assurance activities performed in these enterprises showed that: “testing of web applications” got the highest percentage 65%, “code review” got percentage 43% and finally, “no assurance activities are performed” got percentage 2%. Whereas the results related to determine “who perform assurance activities when developing web applications” showed that “software assurance group” got the highest percentage 52%, “project team” got percentage 37%, and “outside assurance group” got percentage 8%.

4.2 Developers’ Familiarity with Software Processes

Table (2) shows the descriptive statistics of developers’ familiarity with development processes including agile processes. The object-oriented programming got the highest percentage 70%; structured programming got percentage 64%. While XP got small percentage 31% and other agile processes got very small percentages. These results showed that there is a weakness in developers’ familiarity with agile processes. At the same time, the XP is a well known agile process by the developers in these large enterprises.

No.	Development Methodology	Percent
1	Flowcharting	51%
2	Waterfall	63%
3	Structured Programming	64%
4	Structured system analysis and design	50%
5	Information Engineering (IE)	12%
6	Top Down Programming	25%
7	Jackson Structured Programming	2%
8	Dynamic Systems Development Method	4%
9	Object Oriented programming (OOP)	70%
10	Rational Unified Process (RUP)	24%
11	Enterprise Unified Process (EUP)	7%
12	Virtual Finite State Machine (VFSM)	1%
13	Praxis	0%
14	Rapid Application Development (RAD)	33%
15	Spiral RAD	3%
16	Extreme Programming (XP)	31%
17	Agile Methodologies (other than XP)	12%
18	Test Driven Development (TDD)	9%
19	Agile Unified Process (AUP)	7%

Table 2. Developers’ Familiarity with Software Processes

4.3 Software Processes Used by Large Enterprises

Table (3) shows development processes used by these large enterprises including agile processes. The waterfall model got the highest percentage 62%. Object-oriented programming got percentage 51%. Whereas test driven development got percentage 6%, agile unified process got percentage 2% and finally agile methodologies (other than XP) got percentage 1%. These results showed that there is a weakness in adoption of agile processes in these enterprises. At the same time, the XP is a well known agile process used by the developers in these large enterprises.

5. Mixing Agile With Traditional Processes

The software development can not be considered as a defined process because too much change may occur during the development process such as requirements change, technology change and people are added and taken off the team. The focus currently is how to mix agile process with traditional processes for software development. Any organization should: determine the agile properties versus traditional properties; highlight the risks associated with each approach; describe how they would

No.	development methods used by enterprises	Percent
1	Flowcharting	43%
2	Waterfall	62%
3	Structured Programming	48%
4	Structured systems analysis and design	41%
5	Information Engineering (IE/IEM)	14%
6	Top Down Programming	29%
7	Jackson Structured Programming	0%
8	Dynamic Systems Development Method	6%
9	Object Oriented programming (OOP)	51%
10	Rational Unified Process (RUP)	6%
11	Enterprise Unified Process (EUP)	3%
12	Virtual Finite State Machine (VFSM)	0%
13	Praxis	0%
14	Rapid Application Development (RAD)	29%
15	Spiral RAD	0%
16	Extreme Programming (XP)	10%
17	Test Driven Development (TDD)	6%
18	Agile Unified Process (AUP)	2%
19	Agile Methodologies (other than XP)	1%

Table 3. Software Processes Used by Large Enterprises

choose which approach to use; be careful for how and when to mix the two approaches and under what circumstances they would mix them [36]. Agile processes promise higher customer satisfaction and a solution to rapidly changing requirements. Traditional approaches promise predictability, stability and high assurance. Both approaches have shortcomings that, if left unaddressed, can lead to project failure [37].

Boehm and Turner [37] listed in their research three categories of risks that can affect the choice of using the agile or traditional process such as: risks stemming from agile approach (such as scalability, criticality, design simplicity and staff skills); risks stemming from traditional approach (such as emerging requirements, constant change and need for rapid results and staff skills); and finally, general environmental risks (such as technology uncertainties, diverse stakeholders and complex systems). They established a framework for examining the characteristics of development approaches according to these three risks.

Today's, the environment of increasing business change requires more adaptable development processes. The software development processes face new risks such as: shortfalls in real-time performance; unrealistic schedules and budgets; and rapidly change in requirements. Therefore, the large organizations focus on development approaches that increase their responsiveness to business change. Agile processes provide organizations with ability to rapidly evolving systems solutions. The large organizations develop their own customized versions of agile processes to suit their development needs and environment.

The process of developing software systems comprises of interactions among three dimensions: stakeholders; guidelines to provide a direction to development effort; and software product generated as a result of this development effort. The project managers of large organizations should map agile practices along these three dimensions by identifying the correct set of metrics. The development team frequently adjusts its focus to best address these dimensions to remain responsive. Agile processes emphasize sufficient planning in each iteration and minimal emphasis on documentation [38]

Many organizations of web applications' development are attempting to utilize both agile and traditional processes. Although, the adoption of agile processes improves the productivity, quality and business satisfaction, there is also necessary need for other methodologies. Vinekar, et. al. [39] suggested that, the agile development requires a suitable organizational culture. It may be difficult to adopt all agile practices in projects that have stable requirements. New organizational structures are needed to sustain these opposing cultures so that organizations can get the full benefits of both agile and traditional development of applications. There is a need for simultaneously managing agile and traditional development of applications.

The client's culture may be the deciding factor in using agile or traditional processes for a project [39]. The development teams in any organization need to find specific project characteristics to determine if they use agile or traditional process, or hybrid of the two to develop projects [33]. Although many of their advocates consider the agile and traditional process polar opposites, synthesizing the two can provide developers with a comprehensive spectrum of tools and options [34].

Finally, as suggested by Boehm and Turner [37], there is a need to balance stability and agility. The analyzing of agile and traditional processes based on: application, management, technical and personnel characteristics. The choice of traditional or agile processes for a given project is largely depend on five factors: the size of project and team; the consequences of failure; the degree of dynamism of the environment; the competence of personnel; and compatibility with the prevailing culture. The traditional development is desirable when the requirements are stable and predictable and when the project is large and complex. Agile development is suitable when there is a high degree of uncertainty and risk in the project, arising from frequently changing requirements. Large and complex projects are more suited to traditional process and making agile development a suboptimal choice. The agile processes need a suitable organizational culture to sustain it, one that is very different from organizational culture needed for traditional processes.

6. Recommendations

Any enterprise faces several challenges when attempt to use both agile and traditional development processes on different projects. This is because the variations between projects' and clients' characteristics and it may be difficult to change the enterprise's characteristics from project to project. These challenges can be viewed at four levels: management and organization culture; people; process; and technology. Agile and traditional processes have conflicting organizational cultures, management and organizational structures. Also the managers face several challenges when they try to use agile processes into traditional organizations.

According to the survey results in these large Jordanian enterprises, we suggested recommendations in the following sub sections that should be considered by these enterprises to: improve their web development processes; to help large enterprises integrate agile processes into their traditional processes; and to overcome as possible as the problems of web development in these enterprises.

6.1 The Management Team

The large number of developers in the large enterprise should be divided into multidisciplinary small teams. Establish a management team to manage the overall development process and all small teams, this leads to more arrangement in web development process in the large enterprise. The management team should include persons which professional in management techniques, web development activities, and have knowledge in web engineering best practices, software process improvement (SPI) and capability maturity model integration (CMMI) key process areas and goals (KPAs).

The main roles of the management team are to focus on: analyzing the main problems of the large web application; determine precisely what the new web application is supposed to do, its objectives, goal, mission and vision; document these aspects in a clear and understandable way; and risk analysis and management.

The management team should also address the organizational policies, human resources, and legal, cultural and social aspects; focus on requirements analysis of business needs and classify requirement into classes; and finally, identify all stakeholder and their roles and get feedback from them during development process.

6.2 Sub Web Applications

The large web application to be developed by large enterprise may consist of huge number of complex interactive activities. This large application should be divided into many small sub web applications according to its size and complexity. The sub web application can be made small enough for few persons to work on (sub team). The large web application can be designed as a set of parallel or concurrent sub applications.

6.3 Mixing Agile XP Process

Adopt XP principles to be used by each sub team in the large enterprise. The agile processes promising priorities to satisfy customers through early and continuous product delivery and allowing requirements changes. To succeed, large enterprises need to change their core business principles and the ways they are conducting business (i.e. change the organization culture). The management team should evaluate risks at each step in development process to determining how much agility is enough.

The adoption of XP agile process by these selected large Jordanian enterprises together with their own traditional process will avoid as possible as many problems obtained from the interviews were carried out with the developers working in these enterprises as follows:

The adoption of XP reusability principle by each sub team lead to: reduce the times of design and code; reduce the development effort; need less number of developers; need less debugging because of use of proven software; and reduce the costs. The adoption of XP customer communications and feed back principles during the overall development process of sub web application will overcome the requirements' misunderstanding and changing problems.

6.4 Software Quality Assurance

The test mechanism and integrating the Software Quality Assurance (SQA) activities for measuring project success during the development process should be adopted by all sub teams in the large enterprise.

Involving the end users as a mechanism for validating the success of a project should be adopted by sub teams in the large enterprise also. The management team should focus on the: quality management and standards; SQA activities; and measuring the success at the end of the development process.

6.5 SPI CMMI

According to the literature, it is clearly that, the CMMI evaluates an organization as a whole and its development processes. The CMMI is a method for software management and an appropriate way to improve processes, whereas traditional and agile processes are methods for software development.

In order to improve the development process and performance of large enterprise, the management team should always: monitor the overall web development progress, sub team activities to be under the conditions of SPI and CMMI KPAs and goals; and also conduct the web engineering best practices.

6.6 Training

We believe that the agile principles will change the way the large enterprises manage their web development process. Thus, we suggested adopting sufficient training for all developers on agile XP principles and do some serious preparation up front by the large enterprise. At the same time, the large enterprise should adopt continues education and training of all developers in large enterprise and teach them how to deal with CMMI KPAs.

7. Evaluation

This section includes the analysis of this work with respect to many points such as: agile processes, type and size of the software, size of the enterprise, SQA and so on. In comparison to our work, the researchers Khalaf and Al-Amawi [35] focused in their research on the impact of using agile process (XP) to enhance the SQ. They proved according to their survey results that when software developers use some XP practices and tools, they will produce a high SQ and apply the customer needs. At the same time, they did not take in their considerations the: type of software (traditional software or web-based application); size of the software (small, medium or large scale software); size of the company; and finally other agile processes (Crystal, Scrum, TDD and so on) in their survey. Instead, they considered the: relationship between developers' gender and XP; developers' Job type; developers' working experience; and communication technique (email or face to face). Table (4) explains the main points in this work (survey in large Jordanian web development enterprises) in comparison to the research [35].

	Survey in Large Jordanian web development enterprises	Survey in Jordanian software development companies[35]
The XP agile process	ã	ã
Other agile processes	ã	×
Software type (traditional or web-based)	ã	×
Application domain	ã	×
Size of software	ã	×
Size of enterprise	ã	×

Software Quality Assurance	ã	ã
Management	ã	×
SPI CMMI	ã	×
Training	ã	×
Relationship between gender and XP	×	ã
Job type	×	ã
Working Experience	×	ã
Communication technique	×	ã
Developers' familiarity with software processes including agile processes	ã	×
Software processes used by these enterprises including XP	ã	ã

Table 4. Main points in this research in comparison to research [35]

8. Conclusion

Developing web applications is significantly different from traditional software development in nature and life cycle and poses many challenges. Therefore, web engineering process model should: include suitable management method; minimize risks; consider requirements' changing; and deliver web application quickly while providing feedback. The large enterprises which develop large web applications suffering from many problems such as: poor management; deadlines; cost and effort estimation; changing requirements; and lack of customer involvement.

There are many problems of traditional processes: the several phases in system development slow down the development process. The second problem is that the requirements specifications are not flexible since it is difficult to identify all customers' requirements at the beginning of the development process. Even if the requirements can be identified, the business world is forever changing. Therefore, the traditional processes have limitations to adopt them in web applications' development because the nature of these applications in rapidly changing requirements.

Many researchers recommend that these problems can be addressed by the adoption of lightweight iterative and incremental approaches such as XP agile process. The XP uses stories, customer communication, refactoring and continuous integration. On the other hand, XP has many limitations such as: XP is used in small to medium size software projects; lack of requirements management and lack of documentation of requirements; and XP is not suitable to adopt it by the large teams. Therefore, and according to above, we suggested integrate and mix agile XP with traditional processes to use them together in large web development enterprises.

There is a lack of surveys in large web development enterprises to examine software processes used by these enterprises; level of adoption of agile processes in these enterprises; and the problems which face these enterprises during the development process.

In this research, a survey has been conducted in five large web development enterprises in Jordan based on questionnaire and interviews. According to the statistical analysis of the survey results, we noticed that there is weakness in adoption of agile processes by these enterprises, and also the XP is the well known agile process by the developers in these enterprises. From the interviews, we noticed that these enterprises faced many problems during the large web development process such as: changing requirements; poor of communication between customers and developers; lack of standardization and refactoring; poor project management; and poor project cost, time, and effort estimation techniques; and lack of professional developers.

Finally, we suggested a set of recommendations that should be considered by these large enterprises: to improve the software processes used by these enterprises; to help these enterprises overcome the challenges of mixing agile XP process with their traditional software processes; and finally to avoid as possible as many web development problems obtained from the interviews in these enterprises.

9. Future Work

According to suggested recommendations, we will propose in a further research, a web engineering process model to overcome limitations of traditional and agile processes; support the suitable development of large web applications within the

time, budget and effort; and finally, overcome challenges of mixing agile with traditional processes in large web development enterprises.

References

- [1] Deshpande, Y., Murugesan, S., Ginige, A., Hansen, S., Schwabe, D., Gaedke, M. and White, B (2002). Web Engineering. *Journal of Web Engineering* 1(1) 003-017.
- [2] Offutt, J (2002). Quality Attributes of Web Software Applications. *IEEE Software* 19 (2) 25-32.
- [3] Murugesan, S., Deshpande, Y., Hansen, S. and Ginige, A (1999). Web Engineering: A New Discipline for Development of Web-Based Systems. In: *Proc. of the 1st International Conference on Software Engineering (ICSE'99), Workshop on Web Engineering*, pages 693-694.
- [4] Ginige, A. and Murugesan, S (2001). The Essence of Web Engineering: Managing the Diversity and Complexity of Web Application Development, *IEEE Multimedia*, IEEE Computer Society DL 8 (2) 22-25.
- [5] Pressman, R (2005). *Software Engineering: A Practitioner's Approach*, 6th edition, McGraw-Hill International Edition.
- [6] Sommerville, I (2004). *Software Engineering* 7th edition, Pearson Addison Wesley, Pearson education limited.
- [7] McDonald, A. and Welland, R (2001). Agile Web Engineering (AWE) Process, Technical Report TR-2001-98, Department of Computing Science, University of Glasgow, Glasgow, Scotland, G12 8QQ, Dec.2001.
- [8] Sampaio, A., Vasconcelos, A. and Sampaio, F (2004). Design and Empirical Evaluation of an Agile Web Engineering Process, 18^o Simpósio Brasileiro de Engenharia de Software.
- [9] Maurer, F. and Martel, S (2002). Extreme Programming Rapid Development for Web-Based Applications, *IEEE Internet Computing Magazine*, IEEE Educational Activities Dept Piscataway, USA 6(1) 86-90.
- [10] Schatz, B. and Abdelshafi, I (2005). Primavera Gets Agile: A Successful Transition to Agile Development, *IEEE Software*, IEEE Computer Society, 22(3).
- [11] Grossman, F., Bergin, J., Leip, D., Merritt, S. and Gotel, O (2004). One XP Experience: Introducing Agile (XP) Software Development into a Culture that is Willing But Not Ready, Grossman F., Bergin J., and IBM Corp.
- [12] Garland, J., and Anthony, R (2003). *Large-Scale Software Architecture, A Practical Guide Using UML*, John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex PO19 8SQ, England.
- [13] Douglas, B (2000). *Software Engineering: A Programming Approach*, 3rd edition, Addison Wesley, Pearson education limited.
- [14] Curtis, B., Krasner, H. and Nell, I (1988). A Field Study of the Software Design Process for Large Systems. *Communications of the ACM* 31 (11).
- [15] Nahrstedt, K. and Balke, W (2005). Towards Building Large Scale Multimedia Systems and Applications: Challenges and Status, *International Multimedia Conference*, In: *Proc. of the 1st ACM International workshop on Multimedia service composition (MSC'05)*, Hilton, Singapore, Session: Keynote, pages 3-10, ACM New York, NY, USA, 11November2005.
- [16] Balasubramanian V., Bashian, A. and Porcher, D (1997). A Large-Scale Hypermedia Application Using Document Management and Web Technologies, In: *Proc. of the 8th ACM conference on Hypertext and Hypermedia*, Southampton, United Kingdom, pages 134-145, ACM New York, NY, USA, 1997.
- [17] Watts, H. and Marc, K (1989). Software Process Modeling: Principles of Entity Process Models, In: *Proc. of the 11th International conference on software engineering*, Pittsburgh, Pennsylvania, United States, pages 331-342, ACM Press New York, USA, 1989.
- [18] Dorothy, G (1992). Incremental Development and Delivery for Large Software Systems, In: *Proc. of the 6th Annual CSR Conference on Large Software Systems*, IEE Colloquium on Software Prototyping and Evolutionary Development, 25(2) pages1-9. London, UK, Nov 1992.
- [19] Howcroft, D. and Carroll, J (2000). A Proposed Methodology for Web Development, *ECIS, a Cyberspace Odyssey*, Hansen, In: *Proc. of the 8th European Conference on Information Systems*, Vienna, July2000.
- [20] Vidgen, R (2002). Constructing a Web Information System Development Methodology. *Information Systems Journal*, Blackwell Science Ltd. 12(3) 247-261.
- [21] Cater-Steel, A., Toleman, M. and Terry R (2008). Addressing the Challenges of Replications of Surveys in Software Engineering Research. *IEEE*.
- [22] Haynes, R. and Friedenberg, M (2006). Best Practices in Agile Software Development, Technical Report No.0018, College of Information Sciences and Technology, The Pennsylvania State University, 19May2006.
- [23] Kent, B., James, G. and Robert M (2001). Manifesto for Agile Software Development, Ward Cunningham, Feb2001, <http://www.agilealliance.org/home>.
- [24] Scott, A (2005) Quality in an Agile World, 34 SQP, 7(4), ASQ.
- [25] Newkirk, J (2002). Introduction to Agile Processes and Extreme Programming, In: *Proc. of ICSE'02*, ACM, Florida, USA, 19-25May 2002.

- [26] Kent, B (1999). *Extreme Programming Explained: Embrace Change*, Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, Reading MA.
- [27] Steinberg, D. and Daniel, P (2004). *Extreme Software Engineering, A Hands-On Approach*, Pearson Education Inc., Upper Saddle River, New Jersey 07458.
- [28] Turk, D., France, R. and Rumpe, B (2005). Research Review: Assumptions Underlying Agile Software-Development Processes, *Journal of Database Management* 16(4) 62-87.
- [29] Mahanti, A (2006). Challenges in Enterprise Adoption of Agile Methods – A Survey. *Journal of Computing and Information Technology* 3 197–206.
- [30] Aoyama, M (1988). Agile Software Process and Its Experience. In *Proc. of the 20th International Conference on Software Engineering*, IEEE, pages 3-12, Kyoto, Japan, 19-25 April 1998.
- [31] Rising, L. and Janoff, S (2000). *AG Communication Systems. The Scrum Software Development Process for Small Teams*, IEEE Software 17(4) 26-32, IEEE Computer Society Press, Los Alamitos, USA
- [32] Cohn, M. and Ford, D (2003). Introducing an Agile Process to an Organization, *IEEE Computer Society*, 36 (6) 74-78, IEEE 74 Computer Introducing an Agile, June 2003.
- [33] Lindvall, et. al. (2002). Empirical Findings in Agile Methods. In: *Proc. of Extreme Programming and Agile Methods. Second XP Universe and First Agile Universe Conference*, Chicago, USA, 24, pages 197-207, 4-7 Aug 2002.
- [34] Boehm, B (2002). Get Ready for Agile Methods with Care, *IEEE 64 Computer*, 35(1) 64-69, Jan 2002.
- [35] Khalaf S. and Al-Amawi A (2009), Enhance Software Development Quality with Agile Processes: A Jordanian Survey, In: *Proc. of International Conference on Information and Communication Systems (iCICS)*, Amman, Jordan, 20-22 Dec 2009.
- [36] Williams, L. and Cockburn, A (2003). Agile Software Development: It's about Feedback and Change. *Computer* 36(6) 39-43 IEEE Computer Society, June 2003.
- [37] Boehm, B. and Turner, R (2003). *Balancing Agility and Discipline: A Guide for the Perplexed*, Boston: Addison-Wesley.
- [38] Meso, P. and Radhika, J (2006). Agile Software Development: Adaptive Systems Principles and Best Practices, *Contemporary Practices in Systems Development, Information Systems Management*, 23 (3) 1 June 2006.
- [39] Vinekar, V., Slinkman, W. and Nerur, S (2006). Can Agile and Traditional Systems Development Approaches Coexist? An Ambidextrous View, *Contemporary Practices. Systems Development, Information Systems Management* 23(1) 31-42.

Author Biographies



Dr. Omaira N. A. Al-Allaf is an Assistant Professor at CIS department, Faculty of Sciences and IT, AL-Zaytoonah Private University of Jordan, Amman-Jordan. She received her B.Sc. in Computer Science from Dept. of CS, Faculty of Sciences, University of Mosul (1990-1994), and received her M.Sc. in Dept. of CS, Faculty of Computer & Mathematical Sciences, University of Mosul (1997-1999), Mosul, Iraq. She worked as Teacher B in Applied Science University (Private)-Amman-Jordan from 2000 to 2004. She received her PhD in CIS from the Department of CIS, The Arab Academy for Banking and Financial Sciences- Amman-Jordan (2004-2008). During 2004-2008, she worked as a part time lecturer in the Arab Academy for Banking and Financial Sciences. Her research interests are in the field of web engineering, software process models, UI design, artificial neural networks and image processing.